

(19) KOREAN INTELLECTUAL PROPERTY OFFICE

KOREAN PATENT ABSTRACTS

(11)Publication number: 1020010089173 A
(43)Date of publication of application: 29.09.2001

(21)Application number: 1020010009614

(22)Date of filing: 26.02.2001

(71)Applicant: INTERNATIONAL BUSINESS
MACHINES CORPORATION.(72)Inventor: IISHIN TAN
YONCHEN RII

(51)Int. Cl. G06F 17 /00

(54) METHOD, DEVICE AND COMPUTER PROGRAM FOR ACCESSING LEGACY APPLICATION ON DISTRIBUTED DATA
PROCESSING SYSTEM

(57) Abstract:

PURPOSE: A method, a device and a computer program for accessing a legacy application are provided to enable a user to access a host application with a browser for accessing the host application without creating an optional application and HTML(Hyper Text Markup Language) document for each screen. CONSTITUTION: The data created from a host application are converted from a host format to a markup language format and form the markup language data. A markup language document is created by applying a style sheet to the markup language data. The markup language document is displayed on a display creating a user interaction. Another display is provided by using a multi-style sheet to the same host application data. In case of receiving the input of the markup language document, the inputted document is converted to a host format and returned to the host application.

copyright KIPO 2002

Legal Status

Date of request for an examination (20010226)

Notification date of refusal decision (00000000)

Final disposal of an application (registration)

Date of final disposal of an application (20040329)

Patent registration number (1004329360000)

Date of registration (20040514)

Number of opposition against the grant of a patent ()

Date of opposition against the grant of a patent (00000000)

Number of trial against decision to refuse ()

Date of requesting trial against decision to refuse ()

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl. G06F 17/00	(11) 공개번호 (43) 공개일자	특2001-0089173 2001년09월29일
(21) 출원번호	10-2001-0009614	
(22) 출원일자	2001년02월26일	
(30) 우선권주장	09/527,901 2000년03월20일 미국(US)	
(71) 출원인	인터내셔널 비즈니스 머신즈 코퍼레이션, 포만 제프리 엘 미국 000-000 미국 10504 뉴욕주 아몬크	
(72) 발명자	리용청 중국 미국노스캐롤라이나27606알레이,올데트리드라이브107 탄이신 미국 미국노스캐롤라이나27615알레이,그레턴플레이스300	
(74) 대리인	함상준	
(77) 심사청구	있음	
(54) 출원명	분산형 데이터 처리 시스템 상에서 래거시 애플리케이션에 액세스를 제공하기 위한 방법, 장치 및 컴퓨터 프로그램제조물	

요약

본 발명은 애플리케이션에 액세스를 제공하는 방법 및 장치에 관한 것이다. 호스트 애플리케이션으로부터 제공받은 데이터는 호스트 형식에서 마크업 언어 형식으로 변환되어 마크업 언어 데이터를 형성한다. 스타일시트는 상기 마크업 언어 데이터에 적용되어 마크업 언어 문서를 생성한다. 이어, 상기 마크업 언어 문서는 디스플레이 상에 표시되고, 그 문서 상에서 사용자 인터렉션이 발생된다. 상기 마크업 언어 문서에 대한 입력을 수신하면 상기 입력은 상기 호스트 형식으로 변환되어 상기 호스트 애플리케이션으로 복귀한다. 이러한 방법으로 래거시 애플리케이션에 액세스를 제공한다.

대표도

도3

색인어

래거시(lagacy), 마크업 언어(markup language), 스타일시트(stylesheet), 애플리케이션, 액세스

명세서

도면의 간단한 설명

도 1은 본 발명을 구현할 수 있는 분산형 데이터 처리 시스템을 도시한다.

도 2는 본 발명의 바람직한 실시형태에 따른 서버로서 구현할 수 있는 데이터 처리시스템의 블록도이다.

도 3은 본 발명을 구현할 수 있는 데이터 처리 시스템을 나타내는 구성도이다.

도 4는 본 발명의 바람직한 실시형태에 따른 래거시 애플리케이션에 대한 디스플레이 에뮬레이션(display emulation)을 제공하는데 사용되는 구성도이다.

도 5는 래거시 화면을 도시한 것이다.

도 6은 본 발명의 바람직한 실시형태에 따른 변환된 스크린 디스플레이이다.

도 7은 본 발명의 바람직한 실시형태에 따른 다른 형태의 변환된 스크린 디스플레이이다.

도 8은 본 발명의 바람직한 실시형태에 따른 호스트 데이터로부터 화면을 생성하는 과정을 나타내는 플로우차트이다.

도 9는 본 발명의 바람직한 실시형태에 따른 호스트 애플리케이션으로 사용자 입력을 되돌려주는 과정을 나타내는 플로우차트이다.

도 10은 본 발명의 바람직한 실시형태에 따른 호스트 포맷과 XML 포맷사이의 데이터스트림(datastream)을 변환하는 과정을 나타내는 플로우차트이다.

도 11은 본 발명의 바람직한 실시형태에 따른 XML 데이터를 HTML 문서로 변환하는 과정을 보이는 플로우차트이다.

* 도면의 주요부분에 대한 부호의 설명 *

102 : 네트워크	104 : 서버
106 : 저장유닛	108,110,112 : 클라이언트
202,204 : 프로세서	206 : 시스템 버스(bus)
312 : SCSI 호스트 버스 어댑터	314 : 확장 버스 인터페이스
400 : 클라이언트	402 : 호스트 애플리케이션
404 : 호스트 변환	406 : XSL 변환
408 : 스타일시트(styleshet)	410 : 웹 서버

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 향상된 데이터 처리 시스템에 관한 것으로, 상세하게는 애플리케이션에 액세스 제공방법 및 장치에 관한 것이다. 보다 상세하게는, 본 산형 데이터 처리 시스템에서 디스플레이 에뮬레이션(display emulation)수단을 이용하여 레거시 애플리케이션(legacy application)에 액세스 하는 방법 및 장치를 제공하는 것이다.

인터넷 네트워크라고도 불리는 인터넷은 데이터 전송용 네트워크로부터 수신용 네트워크(필요에 따라 패킷을 가짐)에 의해 사용되는 프로토콜 로의 데이터 전송 및 메시지 변환을 조작하는 게이트웨이 수단과 연결된 - 이와 달리 구현될 수 있음 - 일련의 컴퓨터 네트워크이다. "인터넷(Int ernet)"의 머리글자를 대문자로 쓰면, 상기 용어는 TCP/IP 프로토콜을 사용하는 게이트웨이와 네트워크의 집합체를 말한다.

상기 인터넷은 정보 및 오락의 원천으로서 문화적으로 정착되었다. 많은 기업에서는 마케팅을 위한 필수적인 일환으로 인터넷 사이트를 만들어 자사가 제공하는 상품 또는 서비스를 고객에게 홍보하거나 브랜드 가치를 발생시키는 다른 정보를 제공하고 있다. 또한, 다수의 연방,주정부 및 지방자치단체의 부서도, 특히, 국제청 또는 국무장관들과 같이, 사회의 전문분야와 실질적인 상호교류가 필수적인 정부부서에서도 정보제공을 위 해 인터넷 사이트를 운영하고 있다. 정보 안내 및/또는 탐색가능한 공공기록의 데이터베이스를 온라인상으로 제공함으로써 운영비를 절감시킬 수 있다. 또한, 인터넷은 상업적인 거래의 매개체로서도 점차 대중화되고 있다.

현재 인터넷으로 데이터를 전송하는 방법 중에서 가장 일반적으로 사용되는 것은 "웹"이라고도 약칭되는 월드 와이드 웹(World Wide Web) 환 경을 채용하는 것이다. 정보전송에 사용되는 파일 전송 프로토콜(File Transfer Protocol; FTP) 및 고퍼(Gopher)와 같은 다른 인터넷 리소스(res ource)도 있지만, 상기 웹만큼 대중적이지는 못하다. 상기 웹 환경에서는, 서버와 클라이언트는 다양한 데이터 파일(예를 들어, 텍스트, 그래픽 정지화상, 오디오, 동화상 비디오 등등)의 전송을 조작하는 공지된 프로토콜인 하이퍼텍스트 전송 프로토콜(Hypertext Transfer Protocol: HTTP)을 이용하여 데이터를 처리한다. 상기 다양한 데이터 파일의 정보는 표준 페이지 기술 언어 즉, 하이퍼텍스트 마크업 랭귀지(Hypertext Markup Language: HTML)로 사용자에게 프리젠테이션하기 위해 포맷되어 있다. 기본적인 프리젠테이션의 포맷뿐만 아니라, HTML을 통해 개발자들이 유니폼 리소스 로케이터(Uniform Resource Locator; URL)에 의해 식별되는 다른 웹 리소스로 "링크(link)"를 지정할 수 있다. URL은 특정 정보 의 통신경로를 지정하는 특정 구문식별자(syntax identifier)이다. "페이지" 또는 "웹 페이지"라 하며 클라이언트에 액세스 가능한 정보의 로직을 력 각각은 URL에 의해 확인된다. 상기 URL은 이러한 정보를 탐색하고 액세스하기 위한 보편적이고 일관된 방법을 제공하며, 반드시 사용자에게 제공되는 않지만 사용자의 웹브라우저에는 거의 제공된다. 브라우저는 URL과 같은 식별자에 의해 식별된 정보를 요청하도록 제시하는 것이 가능한 프로그램이다. 사용자는 콘텐츠의 소스에 액세스하기 위해 그래픽 사용자 인터페이스(GUI)를 통해 브라우저에 도메인 네임을 입력한다. 상기 도메인 네임은 도메인 네임 시스템(domain name system; DNS)에 의해 인터넷 프로토콜(IP) 어드레스로 자동적으로 변환되며, 상기 DN S는 데이터베이스에서 상기 도메인 네임을 탐색하여 사용자에게 의해 입력된 상징적 이름을 IP 주소로 변환하는 역할을 한다.

또한, 인터넷은 브라우저를 통해 애플리케이션에 사용자의 액세스를 제공하는데 널리 사용된다. 일반적으로, '애플릿(applets)'으로 알려진 프로 그램을 통해 웹 상에서 애플리케이션에 대한 액세스를 제공하여, 상기 애플릿은 그 웹 상의 HTML 문서에서 객체로서 구현될 수 있다. 애플릿은 자신이 생성되어 있는 HTML 페이지와 함께 자바를 지원하는 브라우저 내로 쉽게 다운로드될 수 있는 자바 프로그램이다. 이러한 자바 프로그램 은 독립된 네트워크 및 플랫폼(platform)이다. 애플릿은 자신이 어디에서 발생되었는지 또는 자신이 탑재되는 데이터 처리 시스템이 어떤 것이 지와는 상관없이 동일한 방법으로 구동한다.

그러나, 여전히 많은 래거시 애플리케이션이 존재하며, 이들은 웹을 통해 쉽게 액세스되지 않는다. 래거시 애플리케이션은 오랫동안 존재해온 애플리케이션이다. 상기 래거시 애플리케이션이라는 용어는 종종 메인프레임(mainframe) 및 기업 자원 계획(enterprise resource planning: ERP) 애플리케이션라 하기도 한다. 오늘날의 인터넷 세계에서, 실질적으로 웹에 관련되지 않는 것은 종종 래거시 애플리케이션으로 간주된다. 이러 한 애플리케이션은 아직까지 많은 기업들에 의해 사용되고 있으며, 재정정보, 고객목록, 개인기록 및 생산보고 등과 같은 비즈니스 데이터의 엑 세스를 제공한다. 많은 기업들은 웹을 통해 종업원 및 고객들이 이용할 수 있도록 이러한 애플리케이션에 액세스하기를 원한다.

애플리케이션에 액세스를 제공하는 웹서버와 같은 메인프레임 시스템의 사용이 점차 보편화되고 있다. 그러나 이러한 형태의 시스템 상에서 상 기 래거시 애플리케이션은 사용자가 "녹색 스크린"을 통해 상호작용을 하도록 한다. "녹색 스크린"은 어두운 배경에 녹색문자를 사용하는 표시 장치이다. 이러한 애플리케이션들에 대한 액세스의 제공은 터미널 에뮬레이션(terminal emulation)의 사용을 통해 실현되어 왔으며, 상기 터미 널 에뮬레이션을 사용하려면 에뮬레이션 소프트웨어를 설치하거나 클라이언트 상에 애플릿을 다운로드해야 한다. 다른 경우로는, 클라이언트의 브라우저로 전송하기 위해 "녹색 스크린"의 HTML 버전을 제작하였다. HTML 문서는 하나의 애플리케이션의 각각 스크린을 위해 생성된다. 그 러나, 이러한 스크린의 HTML 버전의 생성하는 것은 번거로운 일이며 시간낭비를 초래한다.

따라서, 웹상에서 래거시 애플리케이션을 액세스하기 위한 향상된 방법 및 장치를 제공하는 것이 유익하다.

발명이 이루고자 하는 기술적 과제

본 발명은 애플리케이션을 액세스하는 방법 및 장치를 제공한다. 호스트 애플리케이션으로부터 발생된 데이터는 호스트 형식에서 마크업 언어 형식(markup language format)으로 변환되어 마크업 언어 데이터를 형성한다. 스타일시트(styleshet)를 상기 마크업 언어 데이터에 적용하여 마크업 언어 문서를 생성한다. 그 다음, 상기 마크업 언어 문서는 사용자 인터랙션(user interaction)이 발생하는 디스플레이 상에 표시된다. 다중 스타일시트는 사용하여 동일한 호스트 애플리케이션 데이터를 위해 다른 디스플레이를 제공할 수 있다.

상기 마크업 언어 문서에 대한 입력을 수신하면, 상기 입력은 상기 호스트 형식으로 변환되어 상기 호스트 애플리케이션으로 복귀한다. 이러한 방법으로, 본 발명의 메카니즘은 래거시 애플리케이션에 대한 액세스를 제공한다.

발명의 구성 및 작용

이하, 도면을 참조하면, 도 1은 본 발명이 구현가능한 분산형 데이터 처리 시스템을 도시한 것이다. 분산형 데이터 처리 시스템(100)은 본 발명이 구현가능한 컴퓨터 네트워크이다. 분산형 데이터 처리 시스템(100)은 네트워크를 포함하며, 상기 네트워크는 분산형 데이터 처리 시스템(100) 내에서 함께 연결된 다양한 장치와 컴퓨터 간의 통신링크를 제공하는데 사용되는 매개체이다. 네트워크(102)는 와이어(wire) 또는 섬유 광케이블과 같은 영구적인 연결 또는 전화연결을 통한 임시적 연결을 포함할 수도 있다.

도 1에 도시한 바와 같이 서버(104)는 저장유닛(106)과 함께 네트워크(102)에 연결된다. 또한, 클라이언트(108,110,112)도 네트워크(102)에 연결된다. 상기 클라이언트(108,110,112)의 예로는 개인용 컴퓨터 또는 네트워크 컴퓨터가 있을 수 있다. 애플리케이션을 위한 네트워크 컴퓨터는 네트워크에 연결된 임의의 컴퓨터로서 그 네트워크에 연결된 다른 컴퓨터로부터 프로그램 또는 다른 애플리케이션을 수신한다. 도 1의 서버(104)는 부트파일(boot file), 운영시스템 이미지 및 애플리케이션과 같은 데이터를 클라이언트(108,110,112)에 제공한다. 클라이언트(108,110,112)는 서버(104)에 대한 클라이언트이다. 도 1에는 도시되지 않았으나, 분산형 데이터 처리 시스템(100)은 서버, 클라이언트 및 다른 장치들을 추가로 포함할 수도 있다.

도시된 실시예에서, 분산형 데이터 처리 시스템(100)은 상호 통신하기 위해 TCP/IP 프로토콜을 사용하는 게이트웨이 및 네트워크의 월드와이드 집합체를 나타내는 네트워크(102)를 갖춘 인터넷이다. 데이터 및 메시지를 전송하는 수많은 상업, 정부, 교육 및 다른 컴퓨터 시스템으로 구성되는, 주요 노드(node) 또는 호스트 컴퓨터 간에 고속 데이터 통신라인의 중추망은 인터넷의 핵심이다. 물론, 분산형 데이터 처리 시스템(100)도 인터넷, 근거리 통신망(LAN), 또는 광역 통신망(WAN)과 같은 많은 다른 형태의 네트워크로 구현될 수도 있다. 도 1은 일예이며 본 발명을 구조적으로 한정하는 것은 아니다.

래거시 애플리케이션은 서버(104)상에 위치할 수 있다. 본 실시예에서 상기 래거시 애플리케이션은 클라이언트(108-112)에 "녹색 스크린" 또는 다른 화면을 제공하는 호스트 애플리케이션이다. 도 1의 클라이언트(108-112)와 같은 클라이언트에 의해 상기 래거시 애플리케이션에 액세스하려면 네트워크(102)를 통해 에뮬레이팅 소프트웨어(emulating software)를 설치하거나 에뮬레이터의 애플릿(applet) 버전을 다운로드해야 하는 경우도 있고 그렇지 않은 경우도 있다.

상기 예에서 본 발명의 메카니즘은 확장가능 마크업 언어(extensible markup language: XML) 및 확장가능 스타일시트 언어(extensible stylesheet language: XSL)의 사용에 기초한 HTML 페이지를 생성한다. 특히, 호스트 데이터스트림(datastream)은 XML 형식으로 변환된다. 상기 호스트 데이터스트림은 사용자 입력을 위한 디스플레이를 포함한다. 상기 예에서, XSL 스타일시트를 상기 변환된 데이터에 적용하여 HTML 페이지를 생성한다. 상기 HTML 페이지를 도 1의 클라이언트(108)와 같은 클라이언트 및 디스플레이를 위한 브라우저로 송신한다. 상기 HTML 페이지는 하이퍼텍스트 전송 프로토콜(Hypertext Transfer Protocol: HTTP)을 이용한 연결을 통해 클라이언트(108)로 송신된다. 입력은 클라이언트(108)에 디스플레이된 HTML 페이지와의 사용자 인터랙션에 응답하여 서버(104)에 의해 수신된다. 상기 입력은 HTTP 형식으로 존재한다. 상기 입력은 이후에 래거시 애플리케이션으로 전송될 변환된 입력의 상태를 형성하면서 래거시 애플리케이션에 의해 인지된 형식으로 변환 또는 변형(transform)된다.

도 2를 참조하면, 도 1의 서버(104)와 같이, 본 발명에 따른 바람직한 실시형태에 따라 서버로 구현가능한 데이터 처리 시스템의 블록도가 도시되어 있다. 데이터 처리 시스템(200)은 시스템 버스(206)에 연결된 복수의 프로세서(202,204)를 포함하는 대칭형 멀티프로세서(symetric multiprocessor: SMP)시스템으로 구현될 수 있다. 이와 달리, 하나의 단일형 프로세서 시스템을 채용할 수 있다. 메모리 컨트롤러/캐쉬(208)도 시스템 버스(206)에 연결되며 로컬 메모리(209)에 인터페이스를 제공한다. I/O 버스 브릿지(bus bridge: 210)는 시스템 버스(206)에 연결되며, I/O 버스(212)에 인터페이스를 제공한다. 메모리 컨트롤러/캐쉬(208) 및 I/O 버스 브릿지(210)는 도시한 바와 같이 통합될 수도 있다.

I/O 버스(212)에 연결된 주변요소 상호연결(peripheral component interconnect: PCI) 버스 브릿지(214)는 PCI 로컬 버스(local bus: 216)에 인터페이스를 제공한다. 다수의 모듈이 PCI 버스(216)에 연결될 수 있다. 통상적으로 PCI 버스의 실행은 네 개의 PCI 확장 슬롯 또는 애드-인 연결부(add-in connector)를 지원한다. 도 1에 도시된 네트워크 컴퓨터(108-112)로의 통신링크는 애드-인 보드(add-in boards)를 통해 PCI 로컬 버스(216)에 연결된 모듈(218) 및 네트워크 어댑터(220)를 통해 제공된다.

추가적인 PCI 버스 브릿지(222,224)는 추가적인 PCI 버스(226,228)에 인터페이스를 제공하고 추가적인 모듈 또는 네트워크 어댑터를 지원할 수 있다. 이러한 방식으로 데이터 처리 시스템(200)은 다중 네트워크 컴퓨터에 연결할 수 있다. 도시된 바와 같이, 메모리 매핑형(memory-mapped) 그래픽 어댑터(230) 및 하드 디스크(232)도 직접적이든 간접적이든 I/O 버스(212)에 연결될 수 있다.

당 기술분야의 숙련자라면 도 2의 하드웨어는 변경할 수 있다는 것을 이해할 것이다. 예를 들어, 광디스크 드라이브 및 그와 유사한 장치와 같은 다른 주변장치를 추가하여 사용할 수도 있고, 상기 하드웨어를 대체하여 사용할 수도 있다. 상기 예는 본 발명의 구조를 한정하려는 것이 아니다.

예를 들어, 도 2의 데이터 처리 시스템은 뉴욕 아몬크 소재의 IBM사(International Business Machines Corporation) 제품으로서 향상된 인터랙티브 실행(Advanced Interactive Executive: AIX) 운영시스템을 구동하는 IBM RISC/System 6000 system으로 구현될 수 있다.

상기 예에서 데이터 처리 시스템(200)은 호스트 또는 래거시 형식과 확장가능 마크업 언어(XML)와 같은 마크업 언어 형식간의 데이터를 변환하는데 사용되는 과정을 포함한다. 또한, 스타일시트를 상기 XML로 변환된 데이터에 적용시키는데 사용되는 상기 과정도 데이터 처리 시스템(200) 내에 위치한다.

도 3을 참조하면, 본 발명이 구현가능한 데이터 처리 시스템의 블록도가 도시되어 있다. 데이터 처리 시스템(300)은 클라이언트 컴퓨터의 일예이다. 데이터 처

리 시스템(300)은 주변 요소 상호연결(PCI) 로컬 버스 구조를 구비한다. 상기 도시된 예가 PCI 버스를 채용하지만, 액셀레이트 그래픽 포트(Accelerated Graphics Port; AGP) 및 산업 표준형 구조(Industry Standard Architecture; ISA) 등의 다른 버스구조를 사용할 수도 있다. 프로세서(302) 및 주 기억장치(304)는 PCI 브릿지(308)를 통해 PCI 로컬 버스(306)와 연결된다. 또한, PCI 브릿지(308)는 프로세서(302)를 위한 통합형 메모리 컨트롤러 및 캐시 메모리를 포함할 수도 있다. 구성요소를 직접 상호연결하거나 애드-인 보드를 통해 PCI 로컬 버스(306)에 추가적으로 연결될 수 있다. 상기 도시된 예에서, 근거리 통신망(LAN) 어댑터(310), SCSI 호스트 버스 어댑터(312) 및 확장 버스 인터페이스(314)는 구성요소의 직접 연결을 통해 PCI 로컬 버스(306)에 연결된다. 그러나, 이와 달리, 오디오 어댑터(316), 그래픽 어댑터(318) 및 오디오/비디오 어댑터(319)는 PCI 로컬 버스(306)에 연결된다. 확장 버스 인터페이스(314)는 키보드 및 마우스 어댑터(320), 모뎀(322) 및 추가적인 메모리(324)를 연결한다. 소형 컴퓨터 시스템 인터페이스(Small computer system interface; SCSI) 호스트 버스 어댑터(312)는 하드 디스크 드라이브(326), 테이프 드라이브(328) 및 CD-ROM 드라이브(330)를 연결한다. 통상적인 PCI 로컬 버스의 구현형태는 세개 또는 네개의 PCI 확장 슬롯 또는 애드-인 연결부를 지원할 것이다.

운영시스템은 프로세서(302)상에서 구동하며, 도 3의 데이터 처리 시스템(300) 내의 다양한 구성요소를 조절하고 제어하는데 사용된다. 상기 운영시스템은 마이크로소프트사에서 제작한 윈도우 2000과 같은 상업적으로 유용한 운영시스템일 수 있다. 자바(Java)와 같은 객체지향적 프로그래밍 시스템은 상기 운영시스템에 연결되어 구동될 수 있으며 데이터 처리 시스템(300)상에서 실행되는 자바 프로그램 또는 애플리케이션으로부터 발생된 호출(calls)을 상기 운영시스템으로 전송한다. "자바"는 선 마이크로시스템사(Sun Microsystems, Inc)의 상표이다. 상기 운영시스템, 상기 객체지향적 운영시스템 및 애플리케이션 또는 프로그램의 명령어는 하드 디스크 드라이브(326)와 같은 저장장치에 위치하며, 프로세서(302)에 의해 실행되기 위해 주 기억장치내에 로딩될 수 있다.

당 기술분야의 숙련자는 도 3의 하드웨어가 그 구현형태에 따라 변경될 수 있다는 것을 이해할 것이다. 다른 내부 하드웨어 또는, 플래시 롬(flash ROM)(또는 그와 유사한 비휘발성 기억장치) 또는 광디스크 드라이브 등의 주변장치를 도3에 도시된 하드웨어에 추가하여 사용하거나 그 하드웨어를 대신하여 사용할 수도 있다. 또한, 본 발명의 프로세스는 멀티프로세서 데이터 처리 시스템에 적용될 수도 있다.

예를 들어, 선택적으로 데이터 처리 시스템(300)을 네트워크 컴퓨터로서 구성하면, 상기 데이터 처리 시스템(300)은 도 3에서 선택적 요소를 나타내는 점선으로 표시된 SCSI 호스트 버스 어댑터(312), 하드 디스크 드라이브(326), 테이프 드라이브(328) 및 CD-ROM(330)을 포함하지 않을 수도 있다. 이러한 경우에, 상기 컴퓨터 - 정확히 말하면, 클라이언트 컴퓨터라 함 - 는 LAN 어댑터(310), 모뎀(322) 등의 특정형태의 네트워크 통신 인터페이스를 갖추어야 한다. 다른 예로서, 데이터 처리 시스템(300)은 특정형태의 네트워크 통신 인터페이스를 포함하더라도, 그 네트워크 통신 인터페이스(300)에 의존하지 않고 부팅할 수 있게 구성된 독립시스템일 수 있다. 다른 예로, 데이터 처리 시스템(300)은 개인 정보 단말기(Personal Digital Assistant; PDA) 장치일 수도 있다.

도 3에 도시된 예와 상기 설명한 예들은 본 발명을 구조적으로 한정하려는 의도가 아니다. 예를 들어, 데이터 처리 시스템(300)은 PDA 형태를 가질 수 있을 뿐만 아니라 노트북 컴퓨터 또는 휴대용 컴퓨터일 수도 있다. 또한, 데이터 처리 시스템(300)은 키오스크(kiosk) 또는 웹 장치일 수도 있다.

도4를 참조하면, 본 발명의 바람직한 실시형태에 따라 래거시 애플리케이션에 디스플레이 에뮬레이션을 제공하는 구성요소의 블록도가 도시되어 있다. 본 실시형태에서는 클라이언트(400)는 호스트 애플리케이션(402)에 액세스한다. 이 액세스는 호스트 변환(404), XSL 변환(406), 스타일시트(408) 및 웹서버(410)를 통해 제공된다.

호스트 애플리케이션(402)은 호스트 데이터를 호스트 변환(404)으로 송신한다. 도4의 실시형태에서는 상기 데이터는 호스트 스크린을 포함한다. 상기 호스트 변환(404)은 상기 호스트 스크린을 호스트 형식에서 마크업 언어 형식 - 예를 들어, XML 데이터를 형성하는 XML 형식 - 으로 변환시킨다.

상기 호스트 형식은 IBM사의 S/390 및 AS/400 호스트와 같은 시스템에 의해 단말기로 전송된 호스트 애플리케이션 데이터 형식인 3270, 5250 및 VT와 같은 다양한 형태를 가질 수 있다.

XML은 웹페이지 및 비즈니스 대 비즈니스 문서 상의 데이터 요소를 정하는데 사용된다. XML은 HTML과 유사한 태그(tag) 구조를 사용한다. XML은 그 요소가 포함하는 대상을 정하는 반면에, HTML은 그 요소의 디스플레이 방법을 정한다. HTML은 미리 정해진 태그를 사용하나, XML은 페이지의 개발자가 태그를 정할 수 있다. 그 결과로, 실제 상의 제품, 판매보고 및 총회비와 같은 임의의 데이터 항목을 식별할 수 있으며 웹페이지는 데이터베이스 기록과 같은 기능을 수행할 수 있다. 이와 같이, XML은 데이터를 식별하는 일반적인 방법을 제공한다. 이러한 예에서, 문서 형식 정의(Document type definition; DTD)를 이용하여 상기 호스트 데이터를 XML 형식으로 변환한다. DTD는 표준 문서작성 언어(standard generalized markup language; SGML)의 내용을 기술하는 언어이다. 상기 DTD도 XML과 함께 사용되며, 상기 DTD 정의는 XML의 내부 또는 별도의 파일 내에 내장될 수 있다.

XSL 변환(406)은 XSL 스타일시트(408)를 XML 데이터에 적용하여 HTML 페이지를 생성한다. 상기 예에서 XSL 변환(406)은 자바 서블릿(Java servlet)의 형태를 갖는다. XSL 변환(406)에 의해 생성된 상기 HTML 페이지는 웹서버(410)로 전송되고, 상기 웹서버(410)는 클라이언트(400)로 차례로 전송하여 브라우저에 디스플레이한다. 이러한 HTML 페이지는 호스트 애플리케이션(402)에 의해 생성된 것과 유사한 인터페이스를 제공한다. XML 데이터가 HTML 형식으로 변환될 때에, 호스트 스크린 입력필드(input field)에 관한 정보를 HTML 입력필드에 삽입하여, HTML 페이지에 대한 사용자 입력을 인식하고 대응하는 호스트 스크린 필드로 다시 변환시킨다. 상기한 예에서, 상기 스타일시트는 상기 HTML 형태로 템플릿(template) 구조를 정의한다. 물론, 상기 스타일시트는 구현형태에 따라 HTML과는 다른 문서 형식을 생성할 수도 있다. 또한, 스타일시트 구문을 이용하여 텍스트 요소가 라인 경계에 따라 쉽게 깨지지 않도록 하는 경우에는 자바스트립트를 사용할 수 있다. 이러한 예에서, 자바 스크립트는 XML 변환과 결합하고 자바 스트립트를 이용하여 호스트 필드의 위치를 계산함으로써 HTML에서 상기 호스트 스크린을 렌더링(rendering)할 수도 있다.

상기 HTML페이지에 대한 사용자 입력을 수집하여 HTML 형식으로 웹서버(410)에 다시 전송된다. 따라서, 상기 사용자 입력은 호스트 변환(404)으로 복귀하고, 상기 호스트 변환(404)은 상기 데이터를 호스트 애플리케이션(402)을 위한 호스트 형식으로 변환한다. 이어, 상기 변환된 데이터는 호스트 애플리케이션(402)으로 복귀한다.

스타일시트를 사용함으로써, 본 발명의 메카니즘은 재사용가능한 소프트웨어 구성요소를 제공하며, 상기 소프트웨어 구성요소는 호스트 애플리케이션(402)과 같은 래거시 애플리케이션을 위한 스크린 또는 디스플레이의 에뮬레이션을 제공하는데 요구되는 작업 또는 노력을 줄인다.

도4에 도시된 여러 구성요소 간의 기능을 분할하여 설명하는 것은 본 발명의 설명을 용이하게 하기 위함이지 본 발명을 구조적으로 한정하고자 하는 것이 아니다. 예를 들어, 호스트 변환(404), XSL 변환(406) 및 웹서버(410)는 하나의 애플리케이션으로 결합될 수도 있다. 상기 구성요소는 도2에 도시된 데이터 처리 시스템(200)과 같은 하나의 서버에서 실행될 수도 있다. 이와 달리, 호스트 변환(404) 및 XSL 변환(406)은 웹서버 상에 하나의 애플리케이션으로 결합될 수 있다. 나아가, 호스트 애플리케이션(402)은 호스트 변환(404), XSL 변환(406) 및 웹서버(410)과 동일한 데이터 처리 시스템 상에 위치할 수 있으며, 이와 달리, 호스트 애플리케이션(402)은 메인프레임 컴퓨터와 같은 원격 데이터 처리 시스템 상에 위치할 수도 있다.

도5를 참조하면, 래거시 디스플레이가 도시되어 있다. 디스플레이(500)는 메인프레임 컴퓨터상에서 볼 수 있는 것과 같이 래거시 애플리케이션에 의해 종종 제공되는 "녹색 스트림" 디스플레이의 형태를 나타낸다. 디스플레이(500)에서 볼 수 있는 바와 같이, 상기 디스플레이는 그래픽 형식을 대신하여 텍스트 형식으로 이루어져 있다.

도6을 참조하면, 본 발명에 따른 변환된 스크린을 위한 디스플레이의 바람직한 실시형태가 도시되어 있다. 상기 예에서, 디스플레이(600)는 도5에 도시된 디스플레이(500)와 같은 래거시 디스플레이를 에뮬레이트하기 위해 디스플레이된 HTML문서를 예시하고 있다. 상기 예에서의 XSL 스타일시트와 같은 스타일시트는 디스플레이(500)를 위한 호스트 데이터의 변환을 통해 생성된 XML 데이터에 적용된다.

도7을 참조하면, 본 발명에 따른 다른 형태의 변환된 스크린을 위한 디스플레이의 바람직한 실시형태가 도시되어 있다. 디스플레이(700)는 HTML 문서로 되어있다. 디스플레이(700)는 호스트 데이터로부터 생성되는 다른 형식의 디스플레이에 대한 일예이다. 상기 예에서 디스플레이(700)는 도5의 디스플레이(500)를 위한 호스트 데이터를 변환하여 생성된다. 디스플레이(700)는 사용자가 디버깅(debugging) 작동을 실행함으로써 보다 쉽게 이용될 수 있다. 디스플레이(700)는 도6의 디스플레이(600)를 생성하는데 사용되었던 스타일시트와 다른 스타일시트를 적용하여 생성된다. 상기 스타일시트를 동일 XML 데이터에 적용한다.

따라서, 수많은 다양한 형식들이 단일 호스트 디스플레이로부터 생성될 수 있다. 이러한 다양한 형식은 상기 호스트 데이터를 XML 형식으로 변환하여 발생된 XML 데이터에 다른 스타일시트를 적용함으로써 생성된다. 이와 같은 방법으로, 다른 클라이언트는 동일한 호스트 애플리케이션으로부터 상기 적용된 스타일시트에 따라 다른 디스플레이를 수신할 수 있다.

도8을 참조하면, 본 발명의 바람직한 실시형태에 따라 호스트 데이터로부터 디스플레이를 생성하는 과정을 나타내는 플로우차트이다. 상기 과정은 호스트 스크린을 수신하는 것으로부터 시작된다(단계800). 상기 호스트 스크린의 XML 표현을 구축한다(단계802). 상기 단계(802)는 도4의 호스트 변환(404)과 같은 호스트 변환 유닛에서 구현된다. 상기 예에서, 상기 과정은 자바빈(Java bean)을 사용하여 구현될 수도 있다. 상기 호스트 스크린의 표현을 구축하는데 이용된 상기 과정은 상기 데이터를 호스트 형식으로부터 XML 형식으로 변환하는 단계를 포함한다.

다음으로, XSL 스타일시트를 XML 표현에 적용하여 HTML 페이지를 생성한다(단계 804). 상기 단계(804)는 도 4의 XSL 변환(406)과 같은 XSL 변환 유닛에 의해 실행된다. 상기 스타일시트는 HTML 서식을 위한 템플릿 구조를 정의한다. XML 데이터 내용은 원래의 "녹색 스트림"의 동일한 행과 열의 위치에 따른 최종적인 HTML 페이지상의 위치와 동적으로 관련된다. 자바스트립트는 텍스트 요소가 라인 경계에 따라 쉽게 분해되도록 허용하지 않는 현재 XSL 스타일시트 구문의 단점을 극복하는데 사용될 수 있다.

단계(804)에서 비보호 필드는 텍스트 타입의 서식 입력 요소로 자동적으로 변환되고, 디스플레이될 수 없는 필드는 패스워드(PASSWORD)타입이 된다. 사용자의 기능키 선택은 상기 기능키의 값을 인식하여 대응하는 호스트 스크린 기능키로 다시 변환하도록 숨은(HIDDEN) 타입 입력 요소로 설정된다. 단말기 스크린의 커서 위치도 상기와 동일한 목적으로 숨은 타입 입력으로 설정된다. 또한, 상기 스타일시트의 이름도 운영자가 스크린의 흐름에 따라 스타일시트의 선택을 동적으로 조작할 수 있도록 숨은 타입 입력으로 설정된다. 상기 조작은 다음 스크린으로 원하는 스타일시트의 이름을 현재 스타일시트에 제시함으로써 실행된다. 그 서식은 XML/XSL 프로세싱을 반복적으로 실행하는 서버 상의 서블릿(servlet)으로 복귀위치를 지정한다. 상기 서식 템플릿은 상기 스타일시트에 내장되며, 상기 스타일시트는 XSL 변환 과정에 의해 처리되어 사용자와의 인터랙션(interaction)을 위한 최종 서식을 생성한다. 따라서 상기 HTML 페이지는 디스플레이하기 위해 클라이언트로 송신되며(단계806), 그 후 상기 과정은 종료된다. 상기 단계(806)는 도4의 웹서버(410)와 같은 웹서버 애플리케이션에 의해 실행될 수 있다.

다음으로 도9를 참조하면, 본 발명의 바람직한 실시형태에 따른 사용자 입력을 호스트 애플리케이션으로 복귀시키는 과정을 나타내는 플로우차트가 도시되어 있다. 상기 과정은 HTML 페이지로 사용자 입력을 수신하는 것으로 시작된다(단계 900). 상기 입력은 브라우저에 적용된 사용자 입력으로부터 수신된다. 상기 입력은 서블릿 또는 컴퓨터 그래픽 인터페이스(CGI)에서 수신된다. 서블릿은 웹서버 또는 애플리케이션 서버에서 구동하는 자바 애플리케이션이며, 통상적으로 데이터베이스에 액세스하거나 전자상거래(e-commerce) 프로세싱을 실행하는 서버-사이드 프로세싱을 제공한다. CGI는 디스플레이 및 출력을 위한 장치 독립 그래픽 언어(device independent graphics language)이다. 상기 서블릿 또는 CGI는 도 4의 웹서버(410)에 구현될 수 있다.

그 다음, 상기 사용자 입력은 호스트 형식으로 변환되고(단계 902), 이어, 상기 사용자 입력은 호스트 애플리케이션으로 전송되며(단계 904), 그 후 과정은 종료한다.

도10을 참조하면, 본 발명의 바람직한 실시형태에 따른 호스트 형식과 XML 형식 사이에서 데이터스트림을 변환하는 과정을 나타내는 플로우차트가 도시되어 있다. 도10에 도시된 과정은 도4의 스크린 변환(404)에서 구현될 수도 있다. 특히, 본 예에서 상기 과정은 자바빈으로 구현될 수도 있다.

상기 과정은 호스트로부터 데이터스트림을 수신하는 단계로 시작한다(단계 1000). 상기 데이터스트림은 파스(parse)된다(단계 1002). 호스트 스크린을 위한 프리젠테이션 공간을 구축한다(단계 1004). 상기 단계(1004)는 상기 호스트 형식의 데이터를 XML 형식으로 변환한다. 본 예에서 상기 변환은 DTD를 이용하여 구현된다. 상기 프리젠테이션 공간을 구축하는데 있어서, 상기 데이터스트림에 의해 기술된 스크린의 레이아웃(layout)을 사용하여 문자 및 속성을 갖는 스크린에 포함된 현재의 모든 필드에 인-메모리(in-memory) 이미지를 생성한다.

그 다음으로, 상기 프리젠테이션 공간을 XML 형식으로 변환하고(단계 1006), 이어, 상기 XML 형식은 XSL 변환 프로세스와 같은 대기중인 프로그램으로 전송된다. 새로운 호스트 데이터에 대해 상기 프로그램의 리스너(listener)에 통지한다(단계 1008). 이후, 상기 과정에서는 사용자 입력을 기다린다(단계 1010). 상기 과정은 사용자 입력이 수신될 때까지 단계(1010)를 반복한다.

사용자 입력이 수신되면, 상기 호스트 스크린에서 대응 입력영역을 식별한다(단계1012). 사용자 입력을 이용하여 상기 호스트 입력필드에 새로운 값을 설정한다(단계1014). 이어, 상기 사용자 입력 및 상기 기능키는 호스트 애플리케이션으로 전송되며(단계1016), 그 후, 단계(1000)로 복귀한다. 상기 단계(1006)에서, 현재 호스트 애플리케이션으로 전송된 상기 데이터는 상기 호스트 애플리케이션에서 사용되는 형식으로 존재한다.

본 발명에서는, 입력/출력 매핑 메카니즘을 채용하여 호스트 터미널을 HTML로 나타냄으로써 사용자 입력을 원래의 호스트 스크린으로 다시 변환시켜 상기 호스트로 다시 전송할 수 있다. 상기 HTML 입력필드는 호스트 애플리케이션에 사용자 입력을 제공하는데 사용하고, 이는 실제로 호스트 입력필드에 관한 정보를 포함한다. 또한, HTML 단말기 에뮬레이터에 있는 숨은 필드는 커서의 위치 및 사용자에게 의한 기능키 입력을 이동시키기 위해 제공된다.

도11을 참조하면, 본 발명의 바람직한 실시형태에 따른 XML 데이터를 HTML 문서로 변환하는 과정을 나타내는 플로우차트를 도시되어 있다. 상기 과정은 상기 XML 데이터를 수신하는 단계로 시작한다(단계 1100). 이어, 사용자를 식별한다(단계 1102). 상기 식별된 사용자에게 맞는 스타일시트를 선택한다(단계 1104). XSL 스타일시트는 상기 스타일시트 이름의 숨은 타입 입력을 조작함으로써 작동시간에 선택되거나 교체될 수 있다. 또한, IBM 호스트 온 디맨드(Host On-Demand; HOD) 및 호스트 퍼블리셔(Host Publisher) 제품 등에 의해 사용되는 매크로 기술(Macro technology)을 이용하여, 호스트 스크린을 자동적으로 스킵(skip)하고 XSL 스타일시트를 동적으로 스워칭할 수도 있다. 매크로는 호스트 애플리케이션을 소정의 상태로 자동적으로 네비게이트(navigate)하는 사전에 기록된 스크립트이다. 이는 사전에 기록된 호스트 액세스 매크로 이름을 위한 다른 숨은 입력 타입에 의해 제공되며, 다음 스타일시트를 적용하기 전에 우선적으로 상기 서블릿이 상기 매크로를 작동하도록 할 것이다.

상기 선택된 스타일시트는 XML 데이터에 적용된다(단계 1106). 이후, 상기 과정은 종료된다. 이러한 방식으로 본 발명을 이용하여 호스트 단말기 에뮬레이션 커스터마이제이션(customization)을 용이하게 실행할 수 있다.

이와 같이, 본 발명은 "녹색 스크린" 디스플레이 시스템을 갖는 애플리케이션과 관련된 문제 및 상기 디스플레이가 지원하기 어려운 인터넷과 같은 네트워크상의 사용과 관련된 문제를 극복한다. 본 발명의 메카니즘은 반드시 특정 애플리케이션을 생성해야 하거나 각 스크린을 위한 HTML 문서를 생성할 필요가 없다. 본 발명은 호스트 데이터를 일반적인 XML 형식으로 변환하고 HTML 문서를 통해 에뮬레이션 인터페이스를 제공하는 XSL 스타일시트를 이용함으로써 상기 잇점을 제공한다. 나아가, 상기 스타일시트는 상기 에뮬레이션 인터페이스를 위해 스크린을 에뮬레이션하도록 로직을 변경할 필요없이 커스터마이징가능한(customizable) 에뮬레이션 인터페이스를 제공하는 메카니즘을 제공한다. 상기 호스트 데이터로부터 생성된 동일한 일반 XML 형식에 상이한 스타일시트를 적용함으로써 특정 사용자의 요구를 만족하도록 상기 인터페이스를 커스터마이징(customize)할 수 있다. 상기 방법에서는, 상기 스크린의 에뮬레이션에 위해 로직을 변경할 필요없이 스타일시트를 대체함으로써 상기 인터페이스를 변경한다. 또한, 상기 방법에서는, 동일한 호스트 애플리케이션에서 발생된 동일한 호스트 데이터를 이용하여, 그 동일한 애플리케이션으로부터 상이한 디스플레이를 상이한 사용자에게 제공할 수 있다.

또한, 본 발명은 사용자가 본 발명을 이용하는 호스트 애플리케이션에 액세스하는 브라우저만 갖추면 되는 잇점을 제공한다. 또한, XSL 스타일시트를 변경함으로써 디스플레이를 쉽게 변화시킬 수 있다.

본 발명을 완전 기능형 데이터 처리 시스템에 관한 내용으로 설명하였으나, 당 기술분야의 숙련자는 본 발명의 과정이 명령어에 대한 컴퓨터 판독가능한 매체의 형태 및 여러 형태로 분산될 수 있으며 본 발명은 이러한 분산을 수행하는데 실제로 사용되는 신호 내포 매체의 지정된 유형과 관련없이 동일하게 적용할 수 있다는 것을 충분히 이해할 것이라는 사실을 주목해야 한다. 컴퓨터 판독가능 매체의 예로는 플로피 디스크, 하드 디스크 드라이브, RAM, CD-ROM 및 CVD-ROM 등의 기록가능형 매체 및 아날로그 통신링크, 무선 주파수 및 광파장 전송과 같은 전송 방식을 사용하는 유선 통신링크 등의 전송형 매체가 있다. 상기 컴퓨터 판독가능 매체는 특정 데이터 처리 시스템에서 실제 사용하기 위해 코드화된 코드형식의 형태를 갖는다.

본 발명의 상세한 설명은 예시 및 설명의 목적으로 제공되었으나, 개시된 형태로 본 발명을 한정하거나 완전하게 설명하려는 것은 아니다. 당 기술분야의 숙련자에 의해 많은 변경 및 개조가 가능하다는 것은 자명하다. 예를 들어, XML 및 HTML이 본 발명에서 설명되었더라도, 다른 문서 언어의 사용도 가능하다.

도시된 실시예가 호스트 형식의 데이터로부터 XML 데이터를 생성하는데 DTD를 사용한다고 예시하고 있지만, 다른 메카니즘을 사용하여 상기 데이터의 전환 또는 변환을 제공할 수도 있다. 예를 들어, XML 문서에 사용된 내용을 정의하는데 XML 스키마(schema)를 사용한다. 상기 XML 스키마는 DTD의 슈퍼셋(superset)이며, 상기 DTD는 표준 SGML 개요이다.

발명의 효과

이상에서 설명한 바와 같이, 본 발명은 "녹색 스크린" 디스플레이 시스템을 가지는 애플리케이션과 관련된 문제 및 상기 디스플레이를 지원하기 어려운 인터넷(Internet)과 같은 네트워크상에서의 상기 애플리케이션 사용과 관련된 문제를 해결할 수 있으며, 또한, 본 발명에서는 사용자가 본 발명을 이용하여 호스트 애플리케이션에 액세스하는 브라우저만 갖추면 되는 잇점을 제공한다.

상기 실시형태는 본 발명의 원리와 실용적인 응용을 가장 잘 설명하고, 의도하는 특정 용도에 적합하도록 다양하게 개조된 여러 실시형태를 위해 당 기술분야의 숙련자가 본 발명을 이해할 수 있도록 선택되어 설명되었다.

(57) 청구의 범위

청구항 1.

데이터 처리 시스템에서 애플리케이션에 액세스를 제공하기 위한 방법에 있어서,

- ① 마크업 언어 데이터(markup language data)를 형성하기 위하여, 상기 애플리케이션으로부터 제공받은 호스트형식의 데이터를 제1 마크업 언어 형식으로 변환하는 단계와,
- ② 마크업 언어 문서를 생성하기 위하여, 스타일시트(styleshet)를 상기 마크업 언어 데이터에 적용하는 단계와,
- ③ 상기 마크업 언어 문서를 표시하는 단계를 포함하는 액세스 제공방법.

청구항 2.

제1항에 있어서,

상기 제1 마크업 언어 형식은 확장가능 마크업 언어(extensible markup language)인 액세스 제공방법.

청구항 3.

제1항에 있어서,

상기 스타일시트는 확장가능 스타일시트 언어(extensible stylesheet language)로 된 스타일시트인 액세스 제공방법.

청구항 4.

제1항에 있어서,

상기 마크업 언어 문서는 하이퍼텍스트 마크업 언어 문서(hypertext markup language document)인 액세스 제공방법.

청구항 5.

제1항에 있어서,

- ① 상기 제1 마크업 언어 형식을 사용하는 입력을 수신하는 단계와,
- ② 호스트 형식 데이터를 형성하기 위하여, 상기 입력을 상기 호스트 형식으로 변환하는 단계와,
- ③ 상기 호스트 형식 데이터를 상기 애플리케이션에 전송하는 단계를 더 포함하는 액세스 제공방법.

청구항 6.

제1항에 있어서,

상기 마크업 언어 문서를 표시하는 단계는 상기 마크업 언어 문서를 상기 데이터 처리 시스템의 클라이언트에 전송하는 단계를 포함하는 액세스 제공방법.

청구항 7.

제1항에 있어서,

상기 마크업 언어 문서는 상기 클라이언트에 있는 브라우저에 표시되는 액세스 제공방법.

청구항 8.

제1항에 있어서,

상기 스타일시트는 에뮬레이션 인터페이스를 위해 스크린을 에뮬레이트(emulate)하기 위한 로직의 변경을 요구하지 않고 상기 에뮬레이션 인터페이스를 커스터마이징(customize)하는데 사용되는 액세스 제공방법.

청구항 9.

데이터 처리 시스템에서 애플리케이션에 액세스를 제공하는 방법에 있어서,

- ① 상기 애플리케이션으로부터 호스트 형식인 디스플레이 정보를 수신하는 단계와,
- ② 확장가능 마크업 언어 데이터를 형성하기 위하여, 상기 디스플레이 정보를 확장가능 마크업 언어 형식으로 변환하는 단계와,
- ③ 디스플레이를 위한 마크업 언어 문서를 생성하기 위하여, 스타일시트를 상기 확장가능 마크업 언어 데이터에 적용하는 단계를 포함하는 액세스 제공방법.

청구항 10.

제9항에 있어서,

상기 하이퍼텍스트 마크업 언어 문서를 클라이언트로 전송하는 단계를 더 포함하는 액세스 제공방법.

청구항 11.

제10항에 있어서,

- ① 상기 클라이언트로 전송된 마크업 언어 문서와의 사용자 인터랙션(interaction)으로부터 생성된 입력을 수신하는 단계와,
- ② 변환된 입력을 형성하기 위하여 상기 입력을 상기 호스트 형식으로 변환하는 단계와,
- ③ 상기 변환된 입력을 상기 애플리케이션으로 전송하는 단계를 더 포함하는 액세스 제공방법.

청구항 12.

제9항에 있어서,

상기 디스플레이 정보는 문서형태의 정의를 이용하여 상기 확장가능 마크업 언어로 변환되는 액세스 제공방법.

청구항 13.

제9항에 있어서,

상기 마크업 언어 문서는 하이퍼텍스트 마크업 언어 문서인 액세스 제공방법.

청구항 14.

제9항에 있어서,

상기 스타일시트는 에뮬레이션 인터페이스를 위해 스크린을 에뮬레이트하기 위한 논리의 변경을 요구하지 않고 상기 에뮬레이션 인터페이스를 커스터마이징하는데 사용되는 액세스 제공방법.

청구항 15.

- ① 버스와,
- ② 네트워크와 통신하기 위해 사용되는 네트워크 어댑터와,
- ③ 상기 버스에 연결되고, 호스트 애플리케이션으로부터 제공되는 데이터스트림을 처리하기 위한 명령어를 포함하는 메모리와,
- ④ 마크업 언어 데이터를 형성하기 위하여 상기 호스트 애플리케이션으로부터 제공되며 호스트형식을 갖는 데이터스트림을 제1 마크업 언어 형식으로 변환하고, 마크업 언어 문서를 생성하기 위하여 스타일시트를 상기 마크업 언어 데이터에 적용하며, 상기 마크업 언어 문서를 클라이언트로 전송하도록, 상기 명령어를 실행하는 프로세서 유닛을 포함하는

데이터 처리 시스템.

청구항 16.

제15항에 있어서,

상기 호스트 애플리케이션은 상기 메모리 내에 위치하는 데이터 처리 시스템.

청구항 17.

제15항에 있어서,

상기 호스트 애플리케이션은 다른 데이터 처리 시스템 상에 위치하며,

상기 데이터스트림은 상기 네트워크 어댑터를 통해 상기 호스트 애플리케이션으로부터 수신되는 데이터 처리 시스템.

청구항 18.

제15항에 있어서,

상기 제1 마크업 언어 형식은 확장가능 마크업 언어인 데이터 처리 시스템.

청구항 19.

제15항에 있어서,

상기 스타일시트는 확장가능 스타일시트 언어로 된 스타일시트인 데이터 처리 시스템.

청구항 20.

제15항에 있어서,

상기 마크업 언어 문서는 하이퍼텍스트 마크업 언어 문서인 데이터 처리 시스템.

청구항 21.

제15항에 있어서,

상기 프로세서 유닛은 적어도 하나의 프로세서를 포함하는 데이터 처리 시스템.

청구항 22.

제15항에 있어서,

상기 프로세서 유닛은 호스트 형식 데이터를 형성하기 위하여, 상기 마크업 언어 문서와의 사용자 인터랙션에 기초한 클라이언트로부터 수신된 입력을 상기 호스트 형식으로 변환하여 상기 호스트 형식 데이터를 상기 호스트 애플리케이션으로 전송하는 데이터 처리 시스템.

청구항 23.

제15항에 있어서,

상기 스타일시트는 에뮬레이션 인터페이스를 위해 스크린을 에뮬레이트하기 위한 논리의 변경을 요구하지 않고 상기 에뮬레이션 인터페이스를 커스터마이징하는데 사용되는 데이터 처리 시스템.

청구항 24.

애플리케이션에 액세스를 제공하는 데이터 처리 시스템에 있어서,

- ① 마크업 언어 데이터를 형성하기 위하여, 상기 애플리케이션으로부터 발생된 호스트 형식인 데이터를 제1 마크업 언어 형식으로 변환하는 변환수단과,
- ② 마크업 언어 문서를 생성하기 위하여, 스타일시트를 상기 마크업 언어 데이터에 적용하는 적용수단과,
- ③ 상기 마크업 언어 문서를 표시하는 표시수단을 포함하는 데이터 처리 시스템.

청구항 25.

제24항에 있어서,

- ① 상기 제1 마크업 언어 형식을 사용하는 입력을 수신하는 수신수단과,
- ② 호스트 형식 데이터를 형성하기 위하여, 상기 입력을 상기 호스트 형식으로 변환하는 변환수단과,
- ③ 상기 호스트 형식 데이터를 상기 호스트 애플리케이션으로 전송하는 전송수단을 더 포함하는 데이터 처리 시스템.

청구항 26.

제24항에 있어서,

상기 마크업 언어 문서를 표시하는 표시수단은 상기 마크업 언어 문서를 상기 데이터 처리 시스템의 클라이언트로 송신하는 송신수단을 포함하는 데이터 처리 시스템.

청구항 27.

애플리케이션에 액세스를 제공하는 데이터 처리 시스템에 있어서,

- ① 애플리케이션으로부터 호스트 형식인 디스플레이 정보를 수신하는 수신수단과,
- ② 확장가능 마크업 언어 데이터를 형성하기 위하여, 상기 디스플레이 정보를 확장가능 마크업 언어 형식으로 변환하는 변환수단과,
- ③ 디스플레이를 위한 마크업 언어 문서를 생성하기 위하여, 스타일시트를 상기 확장가능 마크업 언어 데이터에 적용하는 적용수단을 포함하는 데이터 처리 시스템.

청구항 28.

제27항에 있어서,

클라이언트로 상기 하이퍼텍스트 마크업 언어 문서를 전송하는 전송수단을 더 포함하는 데이터 처리 시스템.

청구항 29.

제28항에 있어서,

- ① 상기 클라이언트로 전송된 상기 마크업 언어 문서와의 사용자 인터랙션으로부터 생성된 입력을 수신하는 수신수단과,
- ② 변환된 입력을 형성하기 위하여 상기 입력을 상기 호스트 형식으로 변환하는 변환수단과,
- ③ 상기 변환된 입력을 상기 애플리케이션으로 전송하는 전송수단을 더 포함하는 데이터 처리 시스템.

청구항 30.

애플리케이션에 액세스를 제공하기 위한 컴퓨터 판독가능 매체에 저장된 컴퓨터 프로그램 제조물에 있어서,

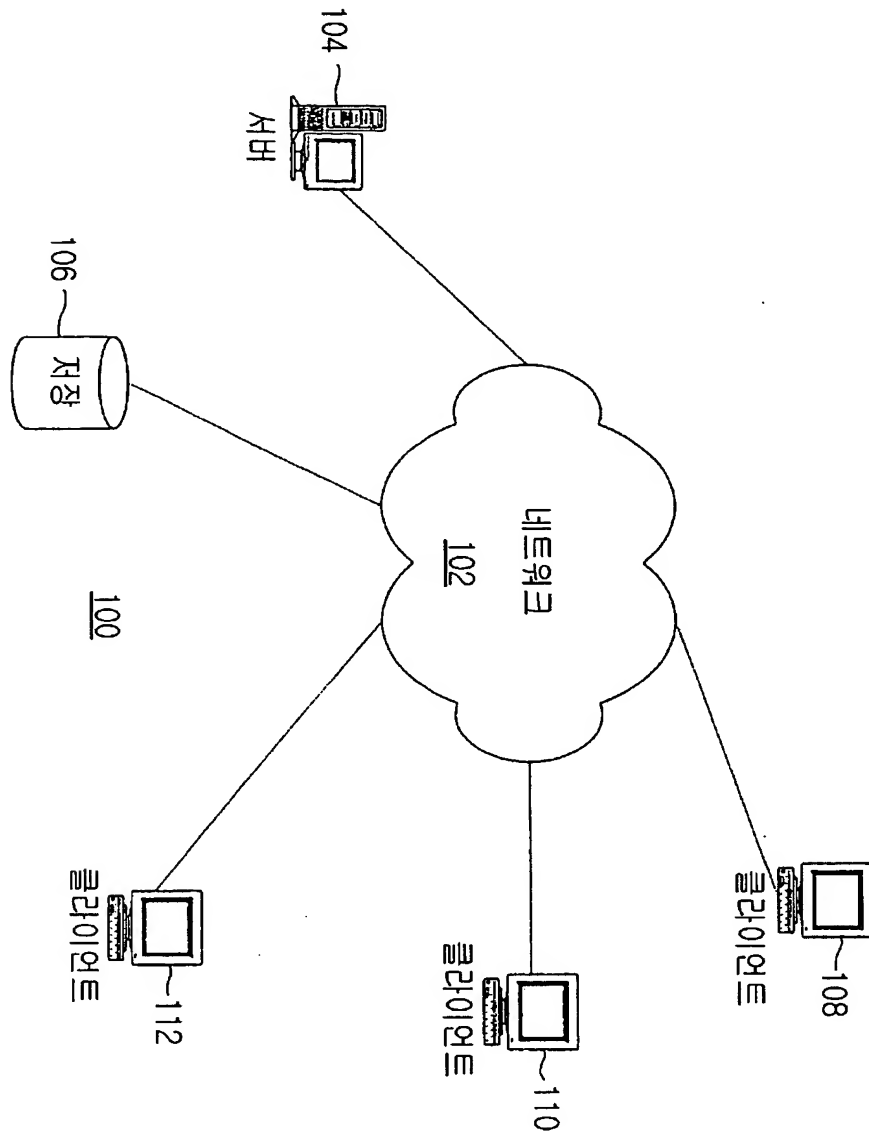
- ① 마크업 언어 데이터를 형성하기 위해, 상기 애플리케이션으로부터 얻은 호스트형식의 데이터를 제1 마크업 언어 형식으로 변환하는 제1 명령어와,
- ② 마크업 언어 문서를 생성하기 위해, 스타일시트를 상기 마크업 언어 데이터에 적용하는 제2 명령어와,
- ③ 상기 마크업 언어 문서를 표시하는 제3 명령어를 포함하는 컴퓨터 프로그램 제조물.

청구항 31.

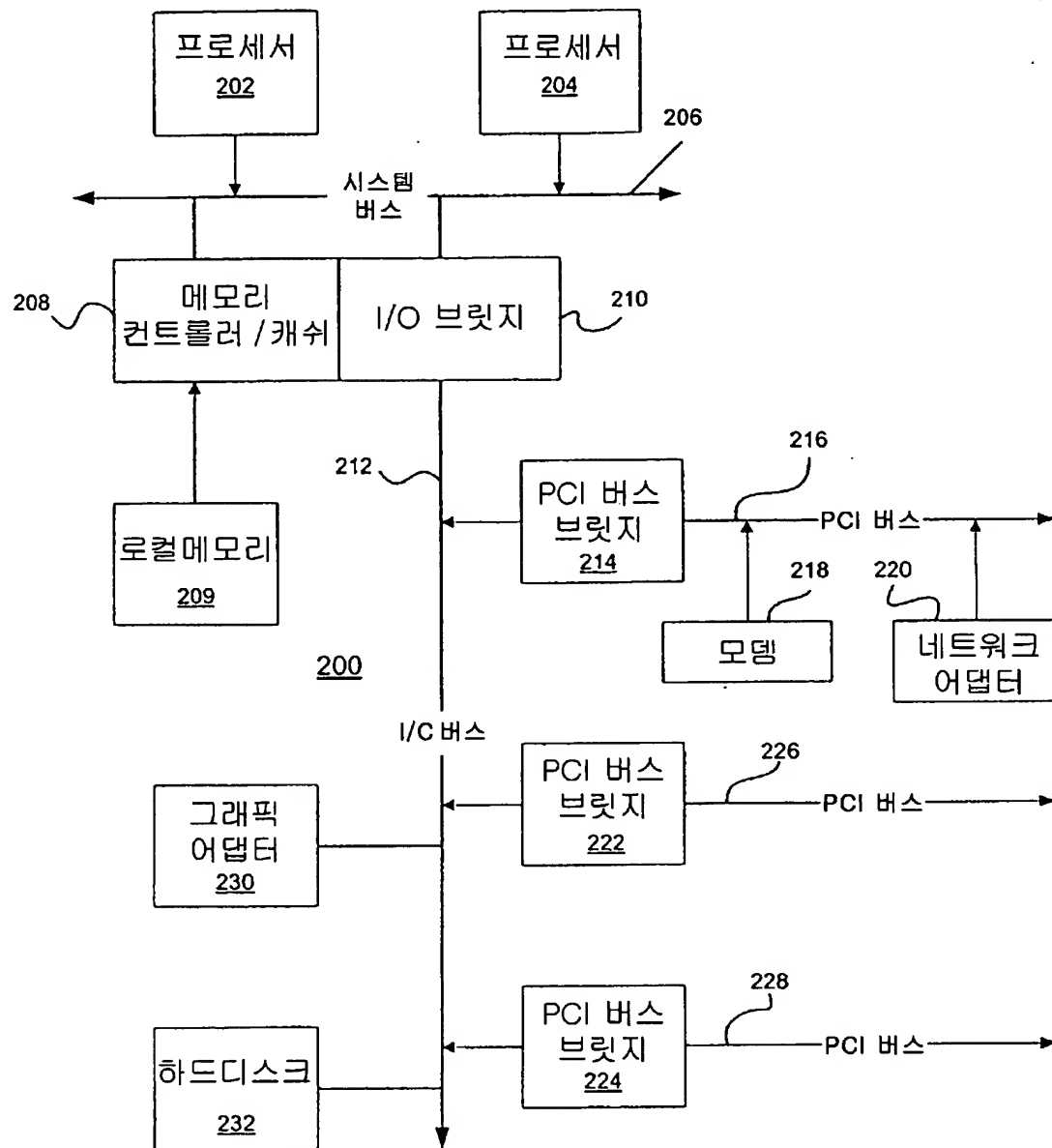
애플리케이션에 액세스를 제공하기 위한 컴퓨터 판독가능 매체에 저장된 컴퓨터 프로그램 제조물에 있어서,

- ① 상기 애플리케이션으로부터 호스트 형식인 디스플레이 정보를 수신하는 제1 명령어와,
- ② 확장가능 마크업 언어 데이터를 형성하기 위해, 상기 디스플레이 정보를 확장가능 마크업 언어 형식으로 변환하는 제2 명령어와,
- ③ 디스플레이를 위한 마크업 언어 문서를 생성하기 위해, 스타일시트를 상기 확장가능 마크업 언어 데이터에 적용하는 제3 명령어를 포함하는 컴퓨터 프로그램 제조물.

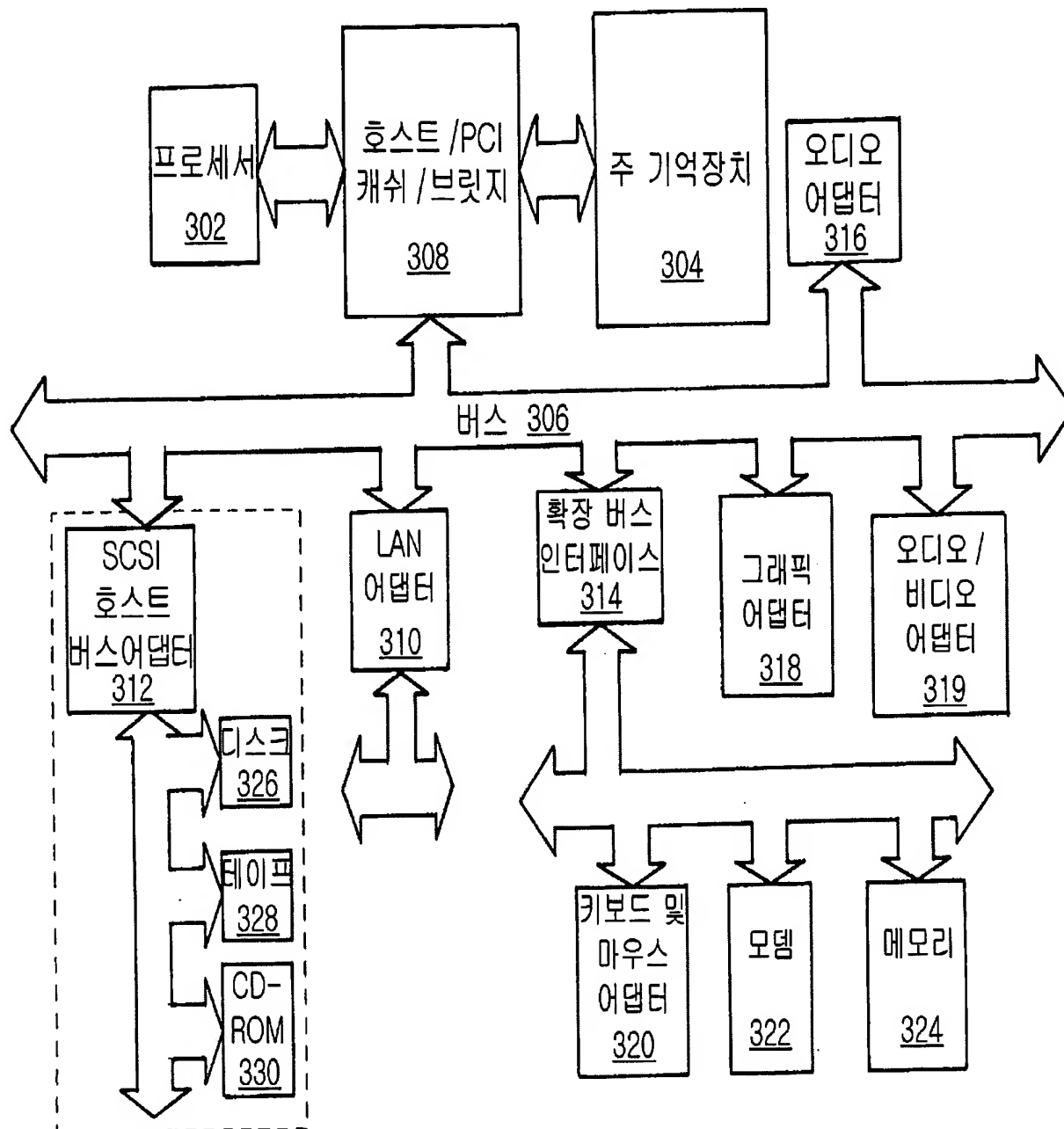
도면



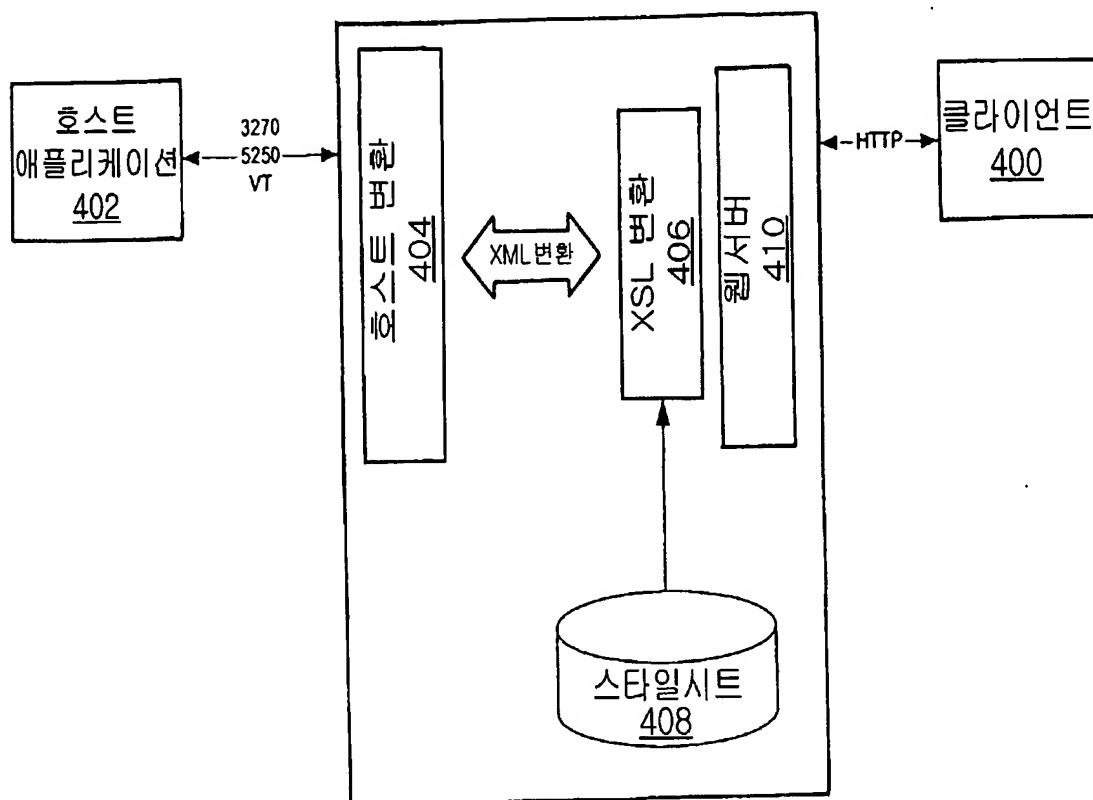
도면 2



도면 3



도면 4



도면 5

Session A - [24 x 80]

File Edit Transfer Appearance Communication Assistant Window Help

VME/ESA ONLINE

WELCOME TO IBM GLOBAL SERVICES

RRR RESEARCH TRIANGLE PARK

R R

RRR III

R R I INFORMATION

R R I

I SSS

III S SYSTEMS

SSS

S

SSS

Customer Service Center: 888-IBM-HELP

USE OF THIS SYSTEM IS FOR IBM MANAGEMENT APPROVED PURPOSES ONLY

Fill in your USERID and PASSWORD (which will not appear) and press ENTER.
If you are already logged on, enter LOGON userid HERE on the COMMAND line.

USERID ==> -

PASSWORD ==>

COMMAND ==>

RUNNING RALVMS

20/016

500

XML 3270 AutoRefresh - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address http://ycli/servlet/xml3270 Go Links

VH/ESA ONLINE

WELCOME TO IBM GLOBAL SERVICES

RRR RESEARCH TRIANGLE PARK

RRR III

RRR I INFORMATION

RRR I

SSS

SSS

SSS

SSS

Customer Service Center 888-IBM-HELP

USE OF THIS SYSTEM IS FOR IBM MANAGEMENT APPROVED PURPOSES ONLY

Fill in your USERID and PASSWORD (which will not appear) and press ENTER.
If you are already logged on, enter LOGON userid HERE on the COMMAND line.

USERID ==>

PASSWORD ==> *****

COMMAND ==>

RUNNING RALVMS

[PF1] [PF2] [PF3] [PF4] [PF5] [PF6] [ENTER] [PA1] [NEWLIN] [INSERT] [ATTN] [REFRESH]

[PF7] [PF8] [PF9] [PF10] [PF11] [PF12] [CLEAR] [PA2] [DELETE] [SYSREQ] [ALTCUR]

Done Local intranet

600

도면 7

XML 3270 AutoRefresh - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address: http://ycli/servlet/xml3270 Go Links

Session Information

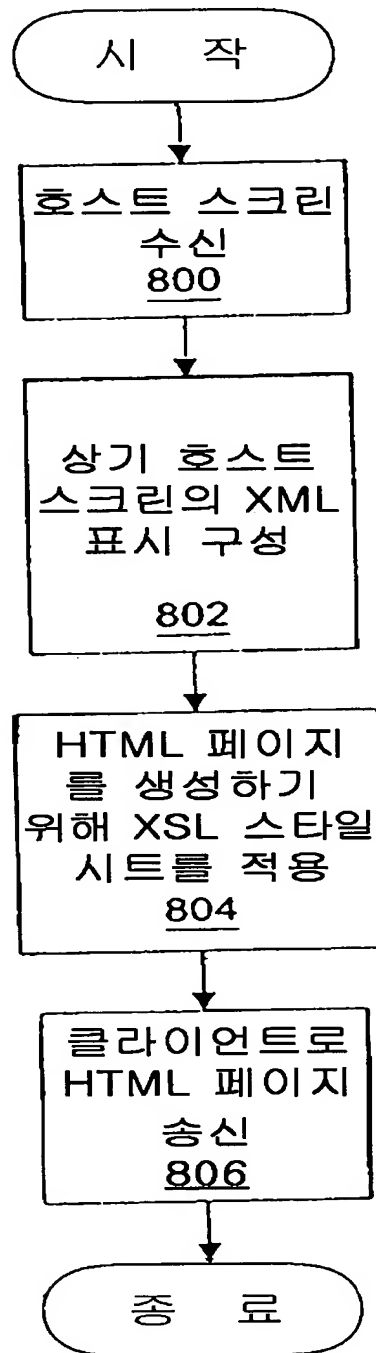
ID	TYPE	DESCRIPTION	HOST	PORT	SIZE
YKTX4DYAAAAACJFHQFQAAA	1		ralvms	23	2

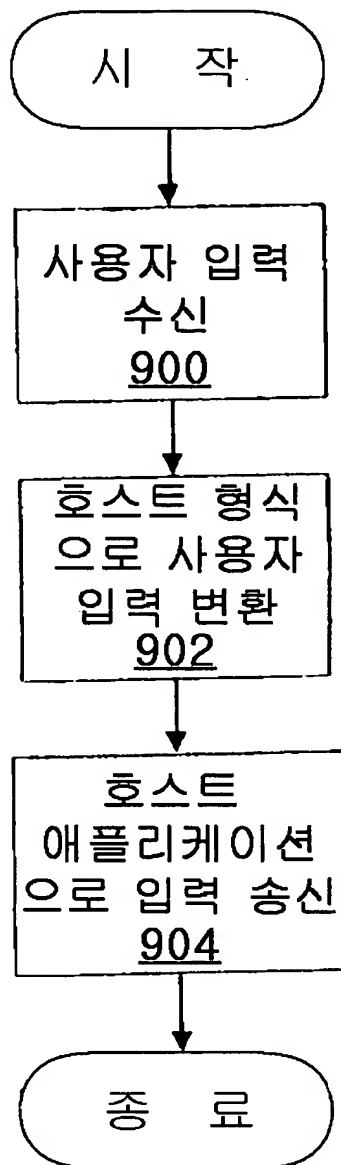
Screen Information

POSITION	LENGTH	PROTECTED	HIDDEN	TEXT
2	13	true	false	VM/ESA ONLINE
16	67	true	false	
84	20	true	false	
105	73	true	false	WELCOME TO IBM GLOBAL SERVICES
179	4	true	false	
184	5	true	false	RRR
190	22	false	false	RESEARCH TRIANGLE PA
213	44	true	false	
258	5	true	false	
264	4	true	false	R R
269	65	true	false	

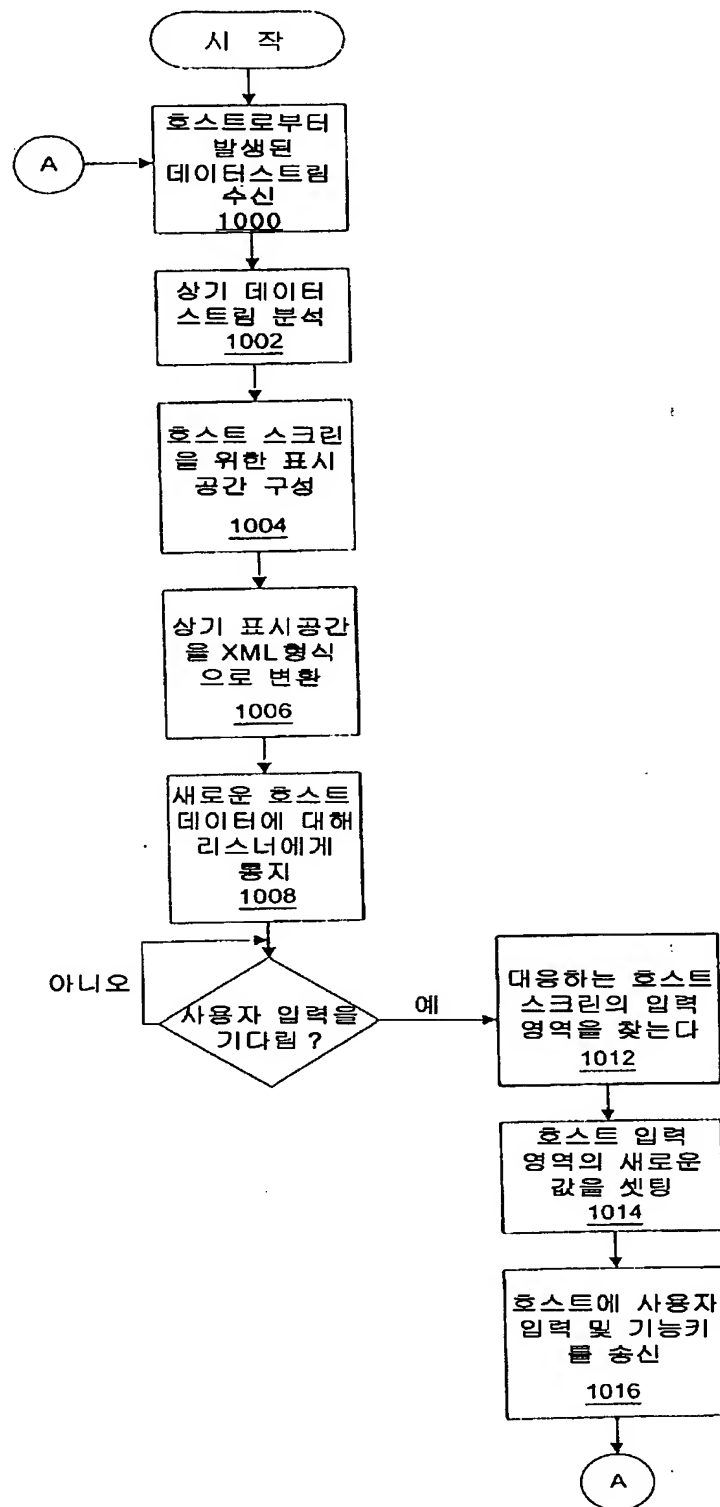
Done Local intranet

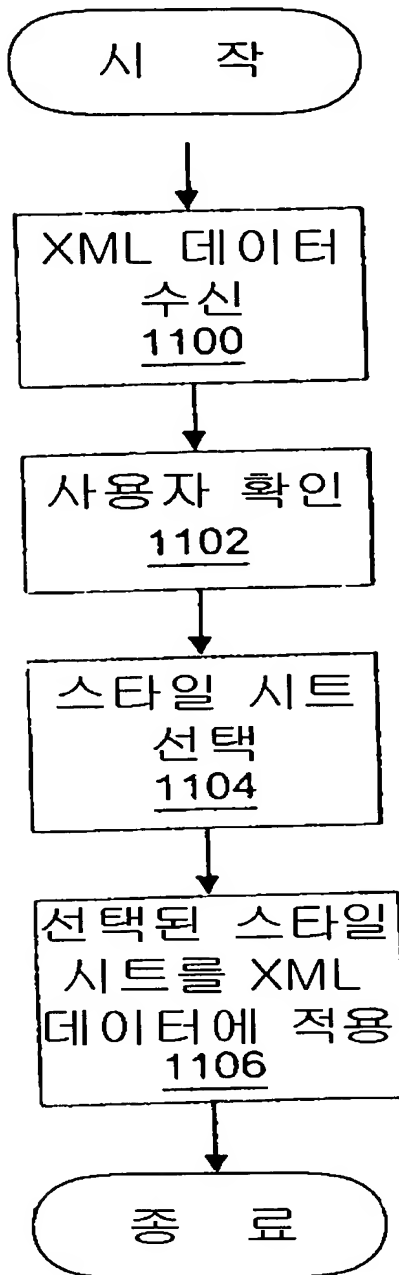
700





도면 10





**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☒ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.